

Memory Integrated Neural Network Accelerators

February 26, 2013

Tarek Taha

**Electrical and Computer Engineering Department,
University of Dayton**

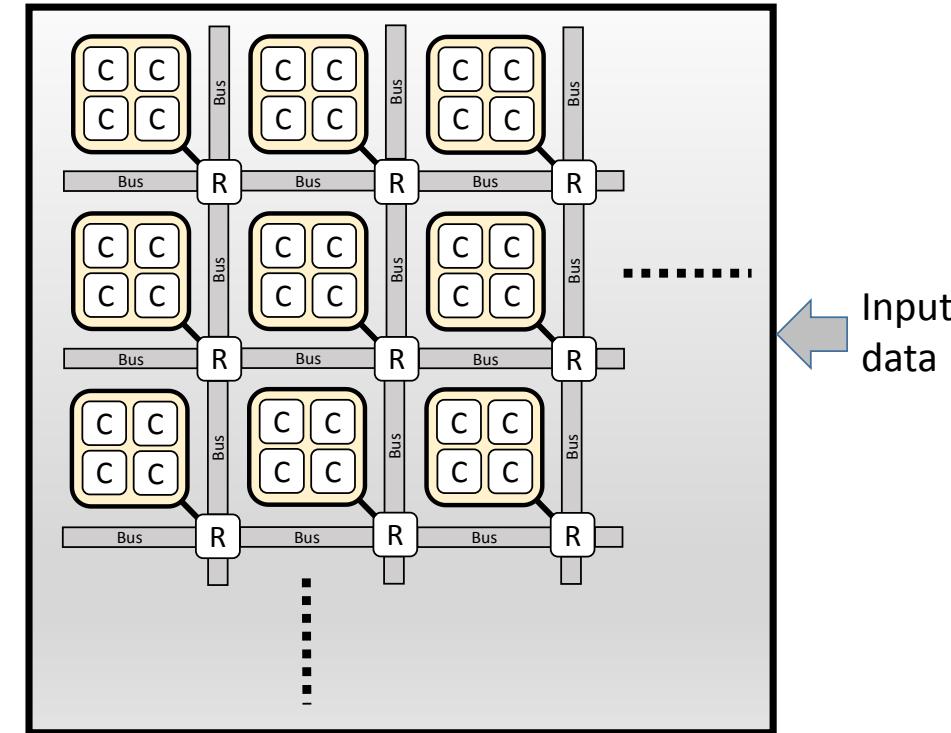


Objective

Design multicore neural network processors

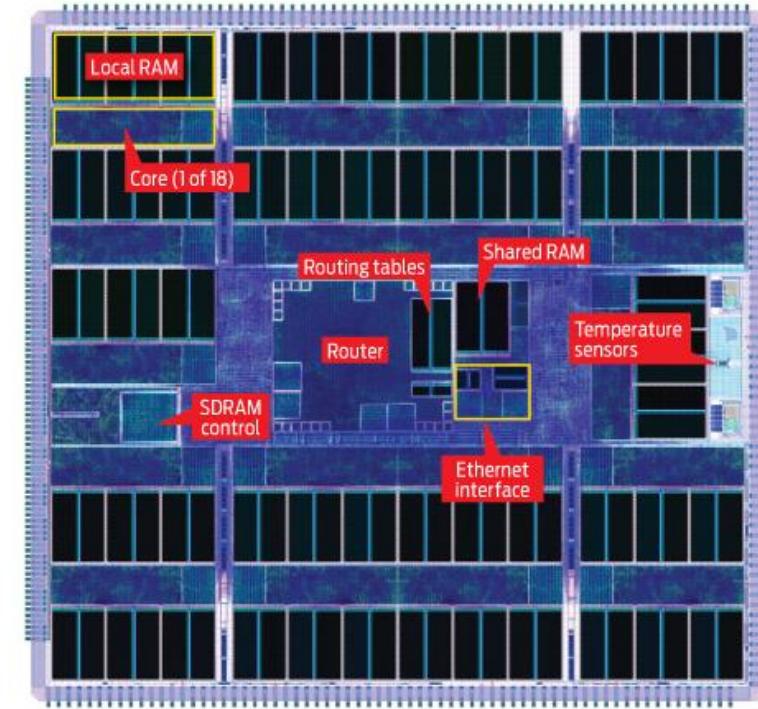
Examine:

- Routing
- Memory options
- Analog and Digital options



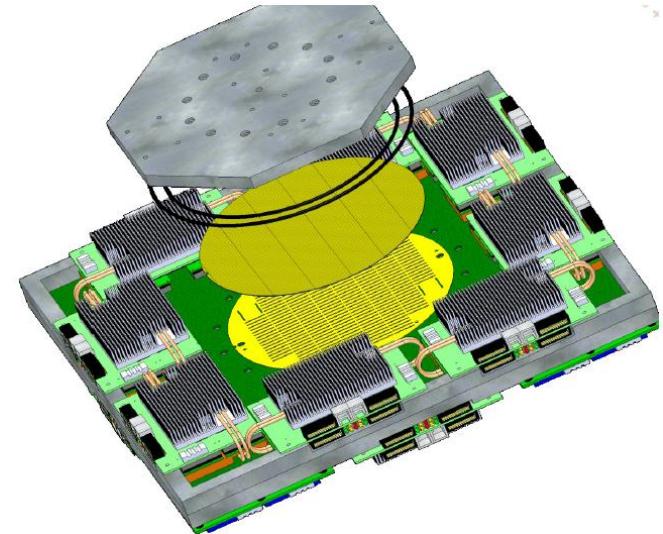
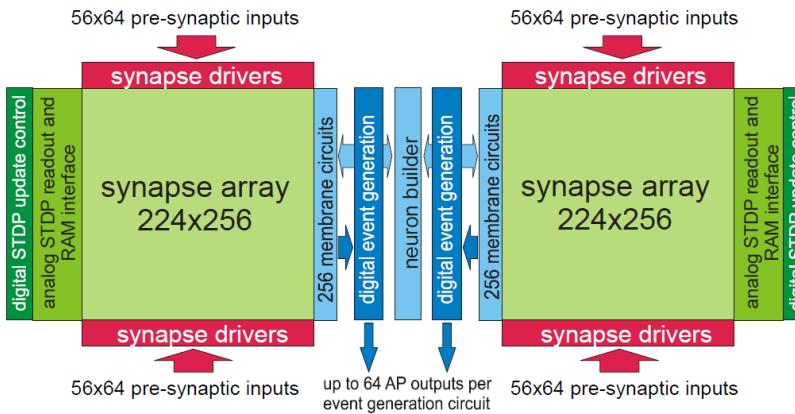
Related Work: SpiNNaker

- ▶ Each chip contains 18 ARM cores and a network router
- ▶ Geared towards brain scale simulation
- ▶ Cores share 128MB SDRAM stacked within the package
- ▶ Multiple chips connected in a 2D toroidal network



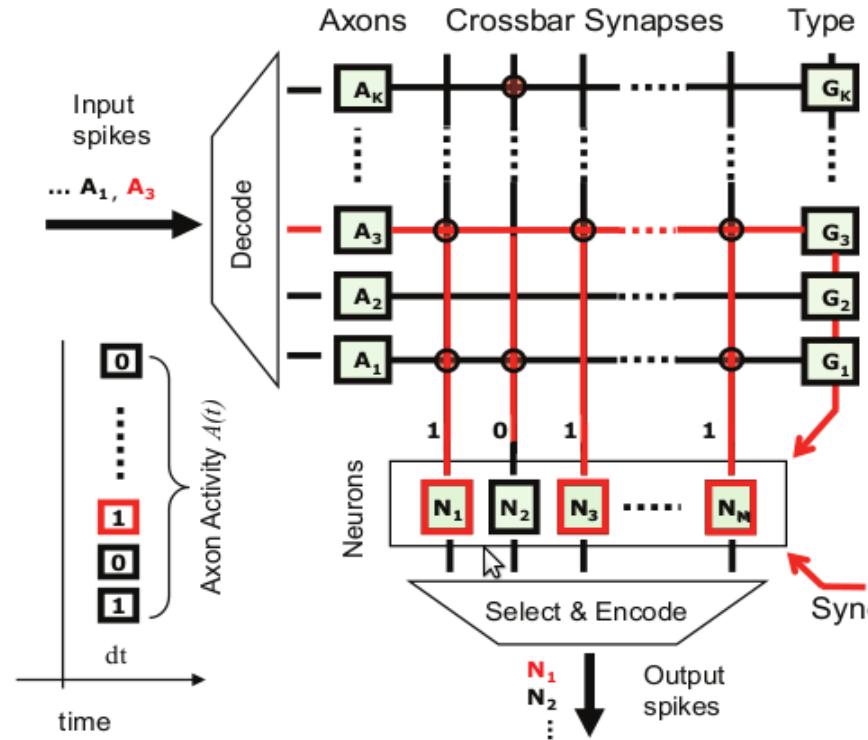
Related Work: FACETS

- ▶ Mixed signal ASIC wafer
- ▶ Geared towards brain scale simulation
- ▶ 200k neurons and 50 million synapses
- ▶ Synaptic weight is represented in a 4-bit SRAM with a 4-bit DAC



Related Work: IBM design

- ▶ Crossbar memory based digital core
- ▶ Each core has capacity to simulate 256 integrate and fire neuron
- ▶ 1024 synapse per neuron
- ▶ One bit per synapse

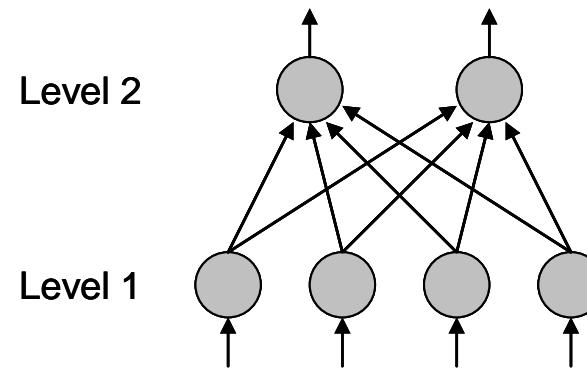


Implementing Synapses

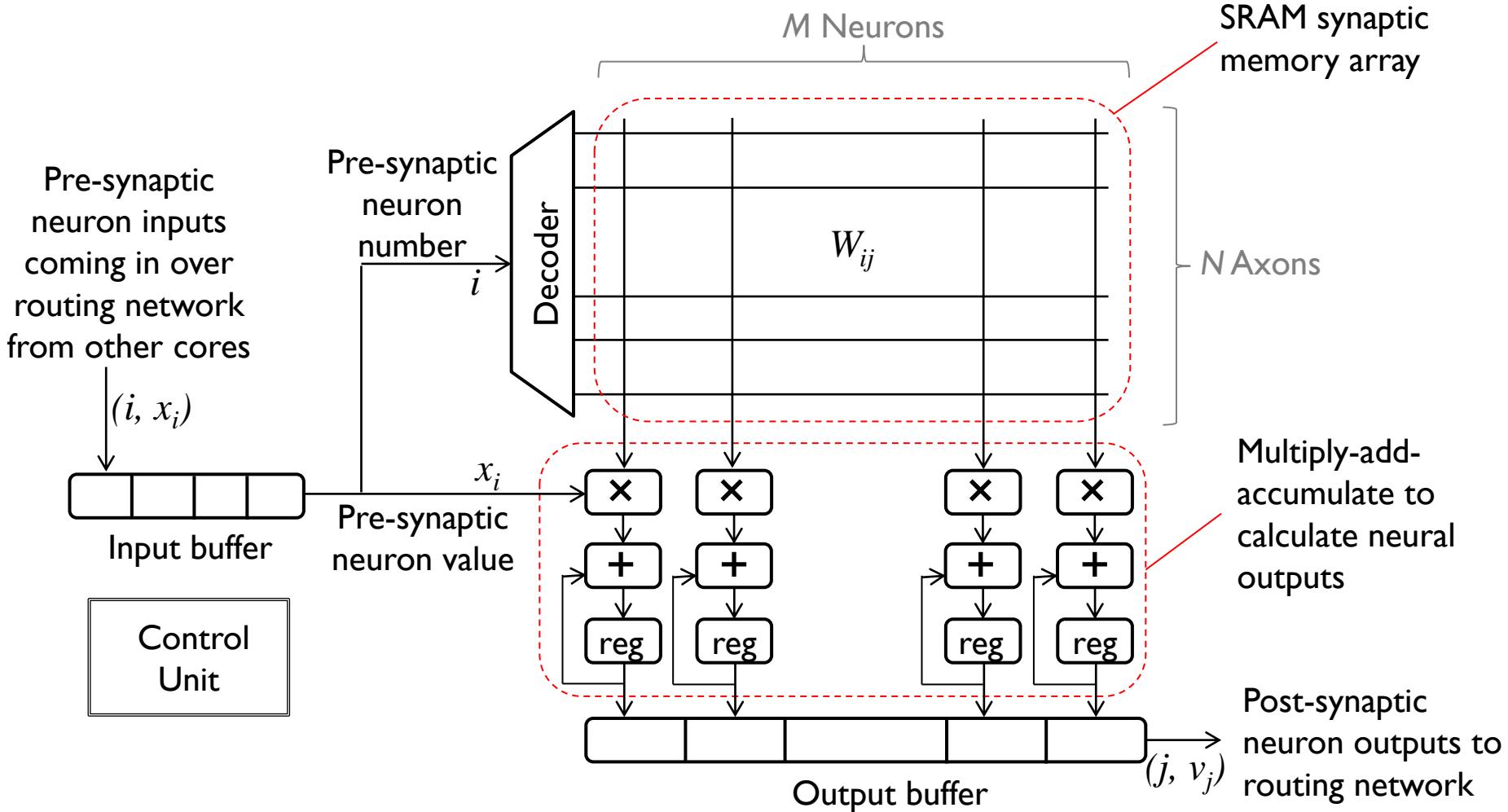
- ▶ Memory elements to implement synapse:

	DRAM	SRAM	Memristor crossbar
Location	Off-chip	On-chip	On-chip
Density (Gb/cm ²)	6.67	0.338	12
Read time	~100	0.3	0.3

- ▶ Memristors can also mimic neural computing circuits



Digital Core Design

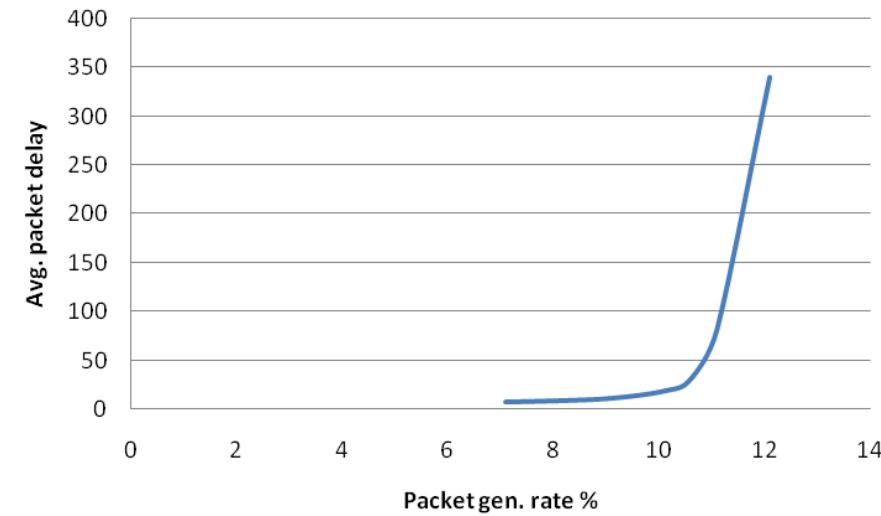
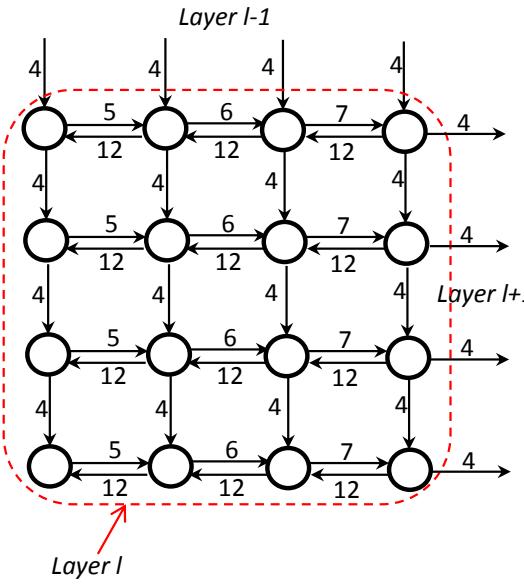


Digital Core Configurations

- ▶ **Configurations:**
 - ▶ 2 bit per synapse, 1 bit per neuron
 - ▶ Multipliers not needed.
 - ▶ 4 bits per synapse
 - ▶ Multiplier and adder needed
- ▶ **Analysis**
 - ▶ SRAM array power and area calculated from CACTI.
 - ▶ Only SRAM array, predecoder, and sense amplifiers considered.
 - ▶ Static and dynamic powers both considered

Routing Analysis

- ▶ Model developed to determine routing network bandwidth requirements
- ▶ Router and link powers calculated from Orion



New Memristor SPICE Model

- ▶ A generalized memristor SPICE model has been developed
 - ▶ Capable of modeling several different published memristor devices for a variety of different voltage inputs
 - ▶ The result of our model has been compared to the published characterization data using a quantifiable error percentage
- ▶ Variation tolerance studies have been performed at the device and circuit level
 - ▶ Layer thickness variation
 - ▶ Variation in state variable dynamics has also been studied
 - ▶ Caused by inconsistent stoichiometric ratio in the metal-oxide layer
 - ▶ Leads to variability in the number oxygen vacancies in each device on a wafer

Model Equations

Current-Voltage (I-V) Relationship $I(t) = \begin{cases} a_1 x(t) \sinh(bV(t)), & V(t) \geq 0 \\ a_2 x(t) \sinh(bV(t)), & V(t) < 0 \end{cases}$

State Variable Motion $\frac{dx}{dt} = \eta g(V(t))f(x(t))$

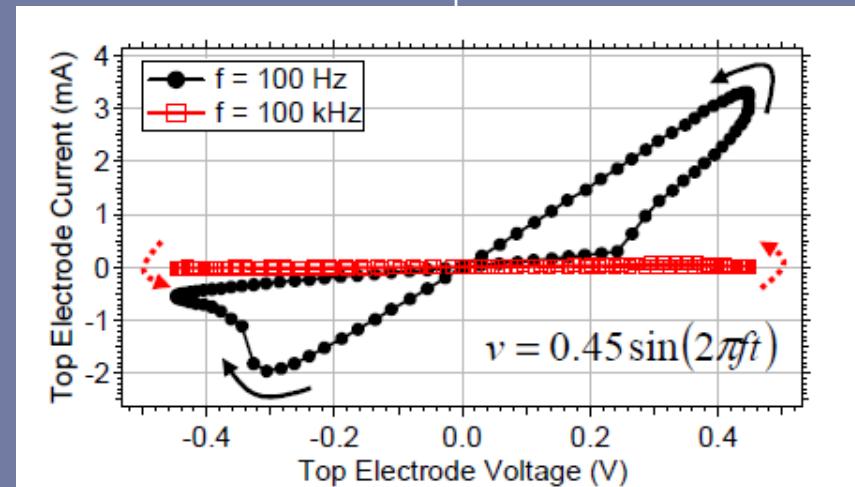
Voltage Threshold $g(V(t)) = \begin{cases} A_p(e^{V(t)} - e^{V_p}), & V(t) > V_p \\ -A_n(e^{-V(t)} - e^{V_n}), & V(t) < -V_n \\ 0, & -V_n \leq V(t) \leq V_p \end{cases}$

Non-Linear Drift $f(x) = \begin{cases} e^{-\alpha_p(x-x_p)} w_p(x, x_p), & x \geq x_p \\ 1, & x < x_p \end{cases}$ $w_p(x, x_p) = \frac{x_p - x}{1 - x_p} + 1$

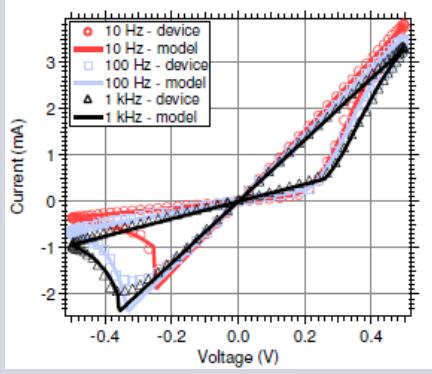
 $f(x) = \begin{cases} e^{\alpha_n(x+x_n-1)} w_n(x, x_n), & x \leq 1 - x_n \\ 1, & x > 1 - x_n \end{cases}$ $w_n(x, x_n) = \frac{x}{1 - x_n}$

Modeling – Sinusoidal Inputs

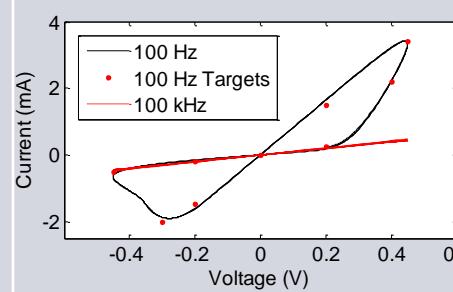
Boise State Device



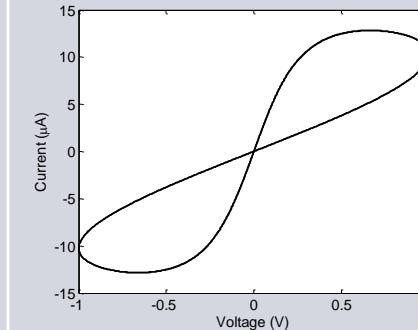
Pino Compact Model



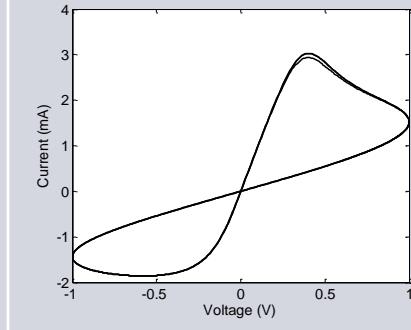
Univ. of Dayton Model



HP Labs Model

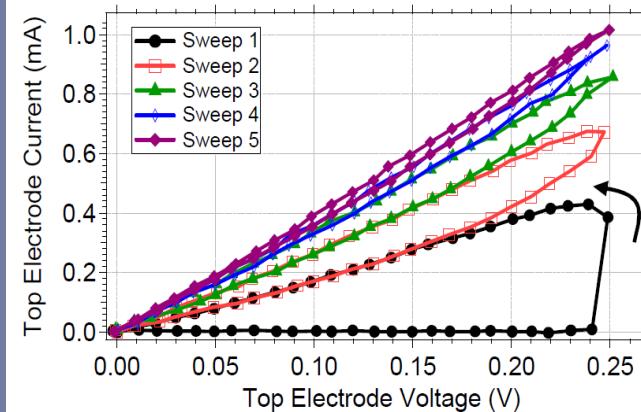


Biolek Model

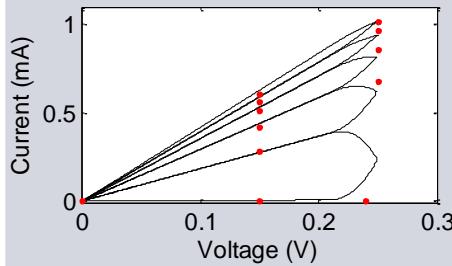


Modeling – Sweeping Inputs

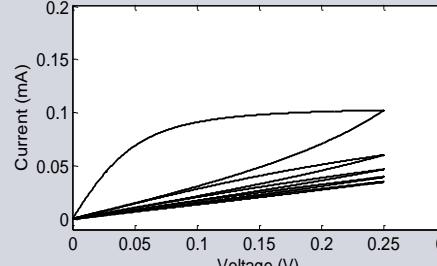
Boise State Device



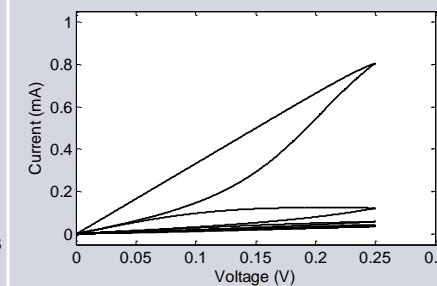
Univ. of Dayton Model*



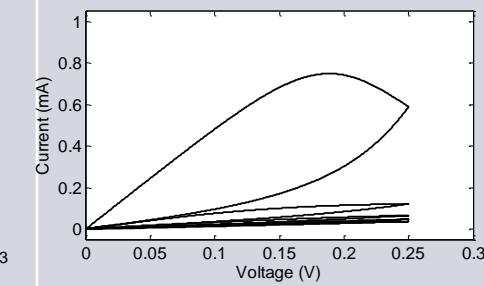
HP Labs Model



Joglekar Model



Biolek Model

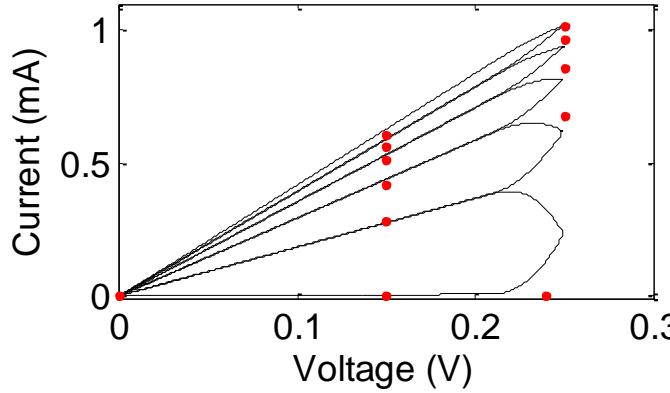
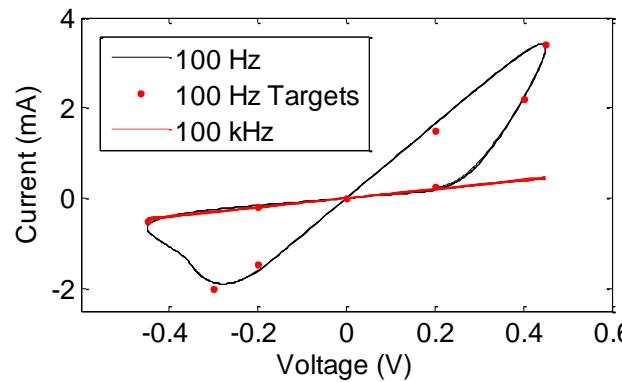


*C. Yakopcic T. M. Taha, G. Subramanyam, R. E. Pino, and S. Rogers, "A Memristor Device Model" *IEEE Electron Device Letters*, 2011.

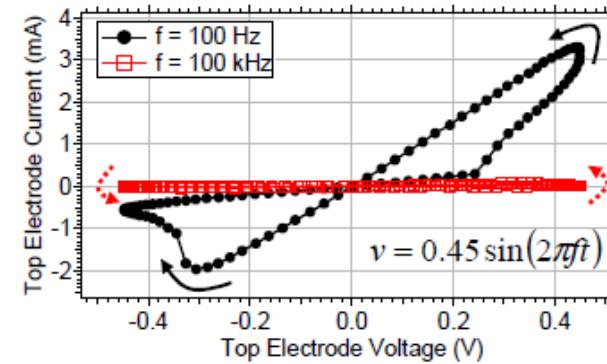
C. Yakopcic, T. M. Taha, G. Subramanyam, E. Shin, P.T. Murray, and S. Rogers, "Memristor-Based Pattern Recognition for Image Processing: an Adaptive Coded Aperture Imaging and Sensing Opportunity," SPIE Adaptive Coded Aperture Imaging and Non-Imaging Sensors. (2010).

UD Model: Boise State Device

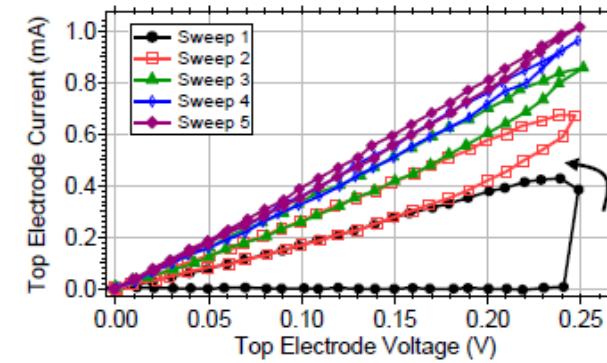
Model result



Data from [I]



Model matches the target data with 6.64% error [I]

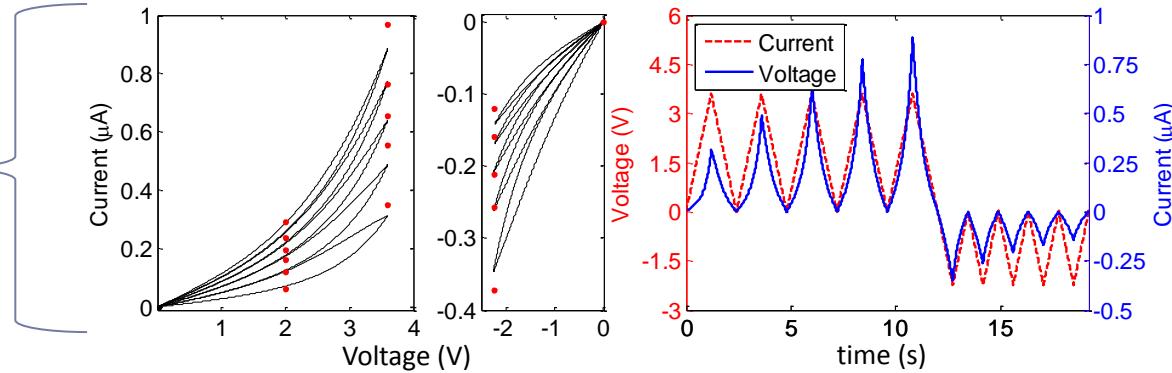


Model matches the target data with 6.66% error [I]

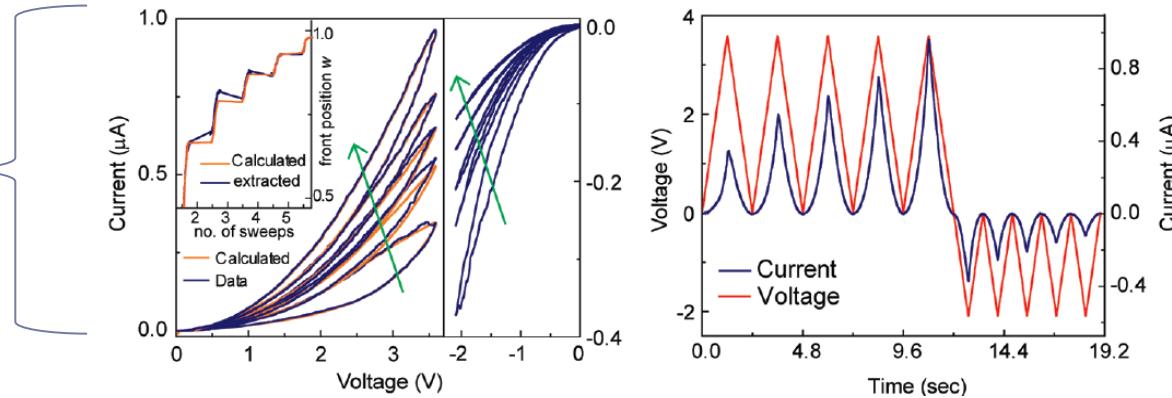
[I] A. S. Oblea, A. Timilsina, D. Moore, and K. A. Campbell, "Silver Chalcogenide Based Memristor Devices," *International Joint Conference on Neural Networks (IJCNN)*, (2010).

UD Model: Univ of MI Device

Our model result



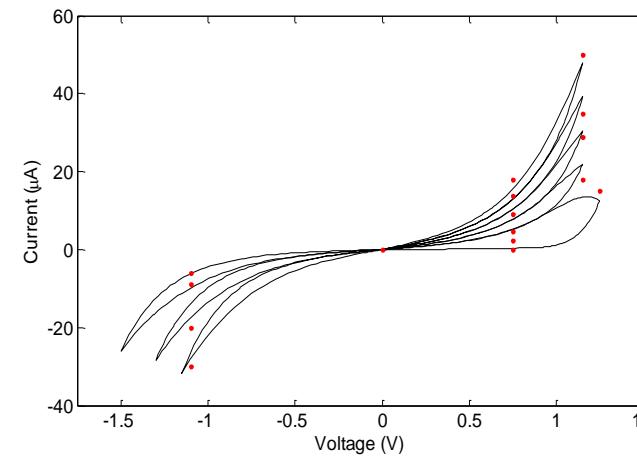
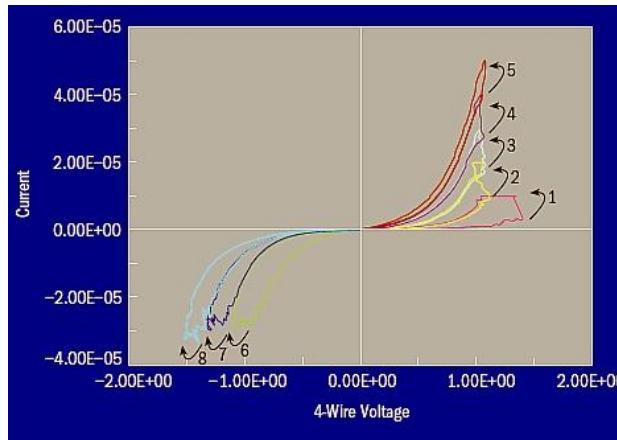
Characterization data from [1]



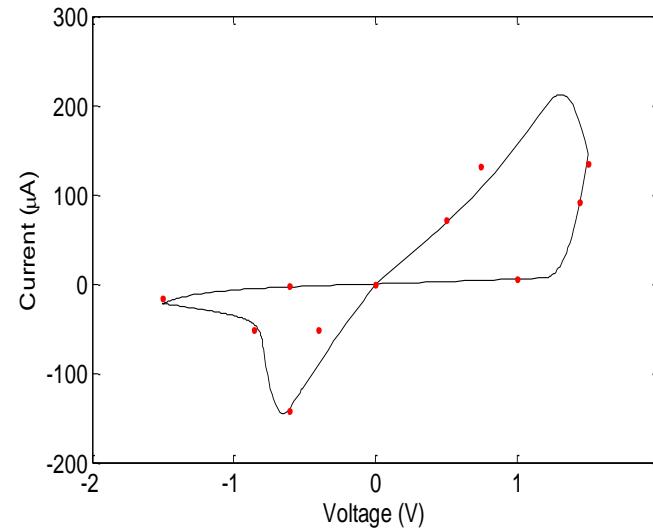
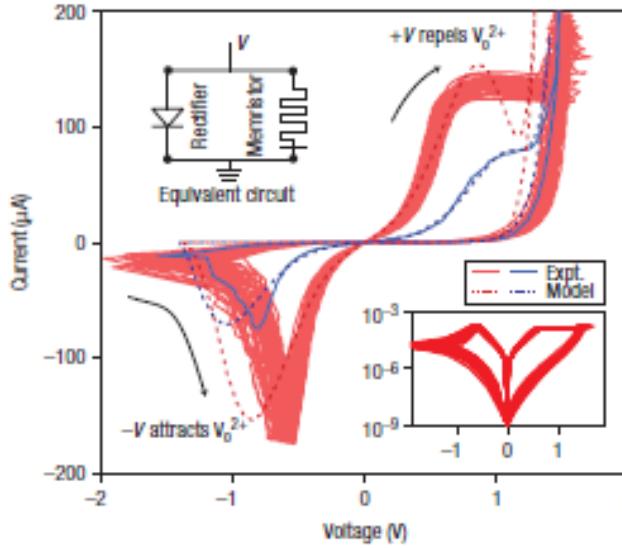
Model matches the target data with 6.21% error

[I] S. H. Jo, T. Chang, I. Ebong, B. B. Bhadviya, P. Mazumder, and W. Lu, "Nanoscale Memristor Device as Synapse in Neuromorphic Systems," *Nano Letters*, 10 (2010).

UD Model: HP Labs Device



Model matches the target data with 11.58% error [1]



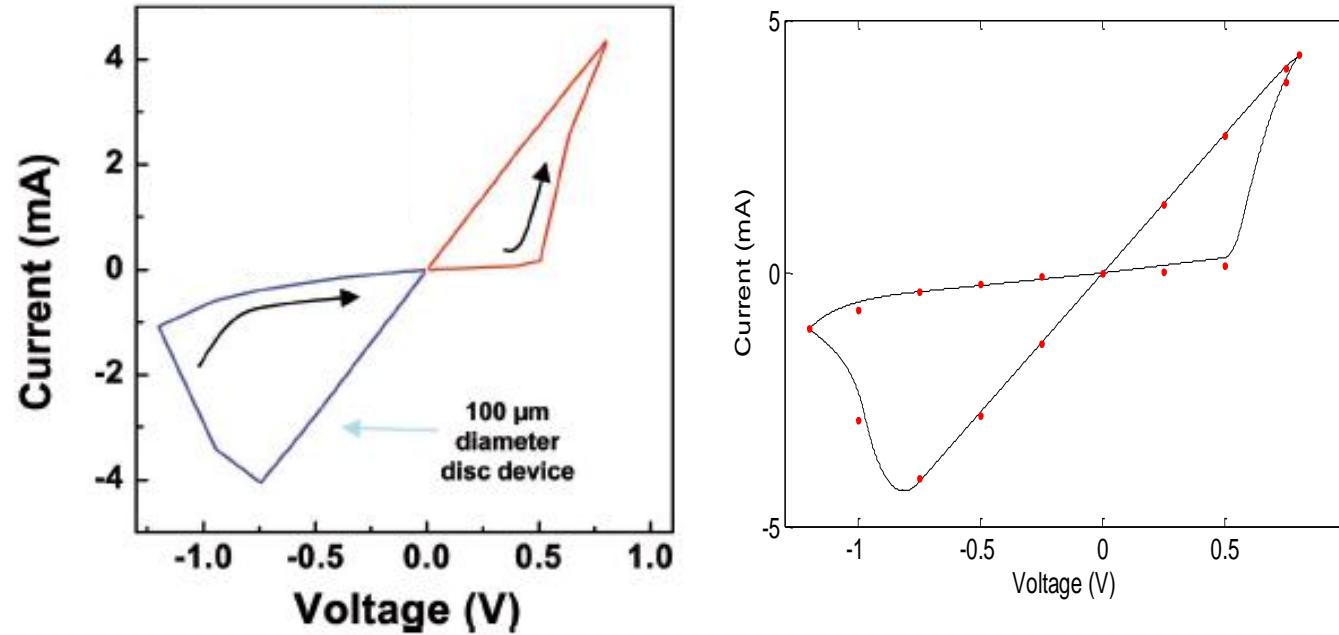
Model matches the target data with 13.6% error (8.72% when not considering the largest outlier [2])

[1] G. S. Snider, "Cortical Computing with Memristive Nanodevices," *SciDAC Review*, (2008).

[2] J. J. Yang, M. D. Pickett, X. Li, D. A. A. Ohlberg, D. R. Stewart and R. S. Williams, "Memristive switching mechanism for metal/oxide/metal nanodevices," *Nature Nanotechnology*, 3, 429–433 (2008).

UD Model: HP Labs TaO_x Device

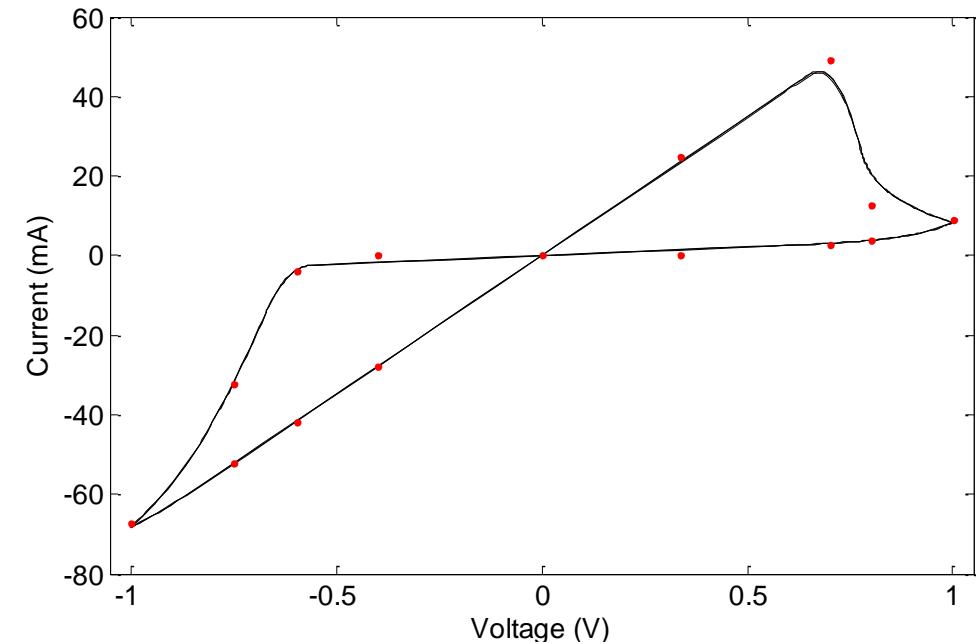
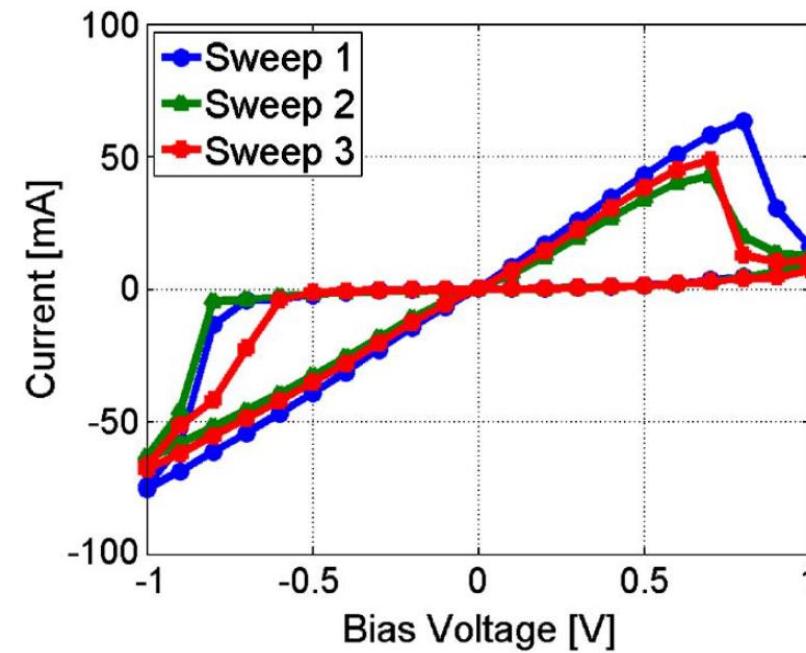
Model matches the target data with 4.60% error



[1] F. Miao, J. P. Strachan, J. J. Yang, M.-X. Zhang, I. Goldfarb, A. C. Torrezan, P. Eschbach, R. D. Kelley, G. Medeiros-Ribeiro, R. S. Williams, 'Anatomy of a Nanoscale Conduction Channel Reveals the Mechanism of a High-Performance Memristor,' Advanced Materials, vol. 23, no. 47, pp. 5633-5640, Nov. 2011.

UD Model: Univ. of Iowa Device

Matches physical characterization with 5.97% error

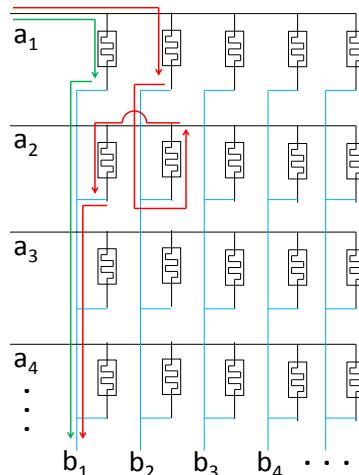


K. Miller, K. S. Nalwa, A. Bergerud, N. M. Neihart, and S. Chaudhary, "Memristive Behavior in Thin Anodic Titania," *IEEE Electron Device Letters* 31(7), (2010).

K. Miller, "Fabrication and modeling of thin-film anodic titania memristors," *Master's Thesis*, Iowa State University, Electrical and Computer Engineering (VLSI), Ames, Iowa, (2010).

Memristor Modeling

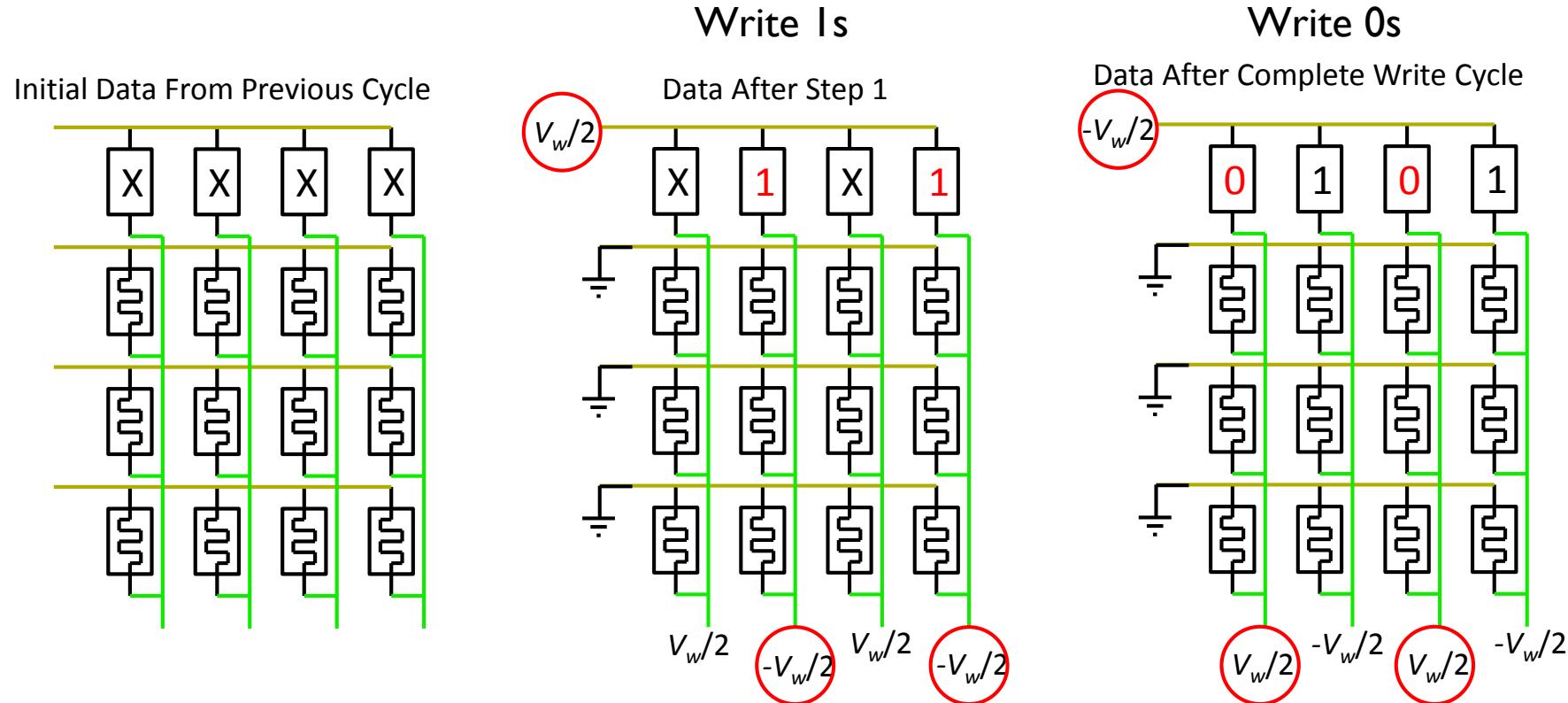
- ▶ Our model is more robust and accurate than other models
- ▶ Results are based on a 16 memristor crossbar circuits with a 10 ns switching time



Measurement	Memristor Models Included in the Study				
	Biolek	Joglekar	Laiho	Chang	Our Model
Time Before Error (μ s)	0.517	0.517	15 (Finished: no error)	0.135	15 (Finished: no error)
Accuracy	Low	Low	Medium	Medium	High
Generality	High	High	Medium	Medium	High

Crossbar Write Technique

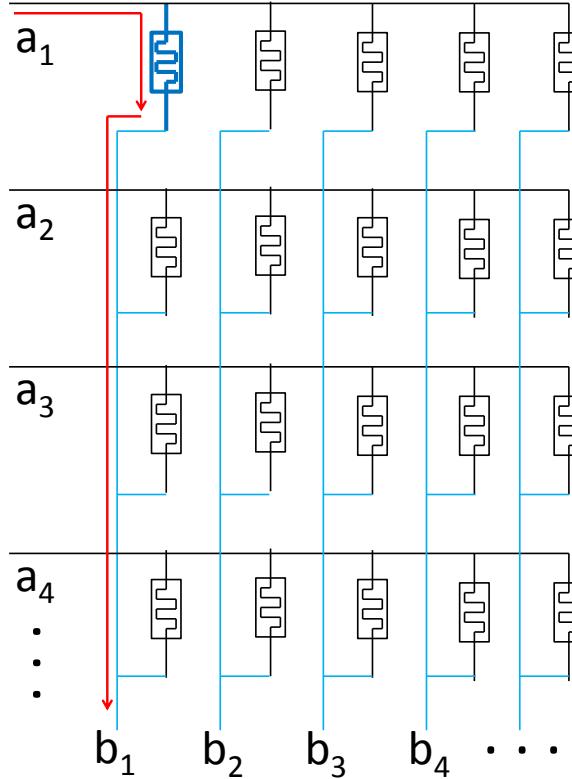
- Write method used stops state change of unselected devices
 - Although this does not solve the problem for reads



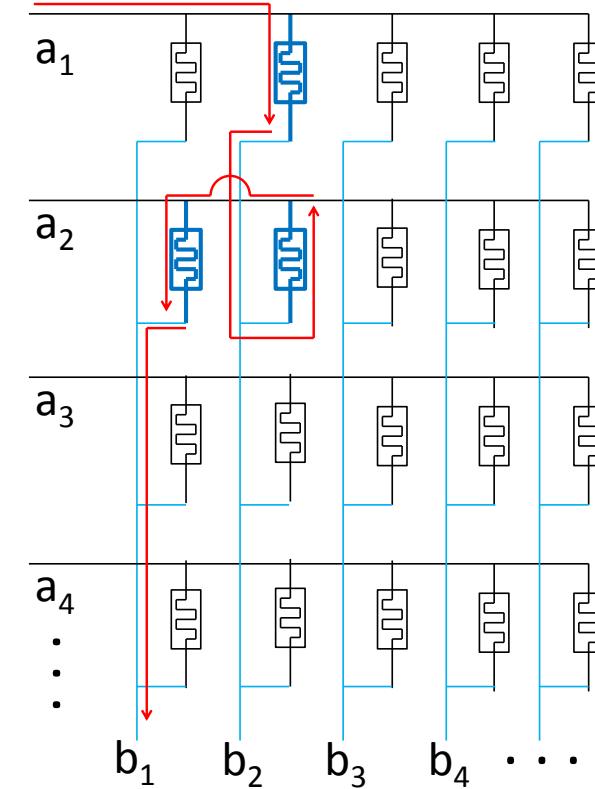
Unwanted Current Paths

- ▶ Alternate current paths in the resistor array consume power and degrade signal.

Correct read between a_1 and b_1



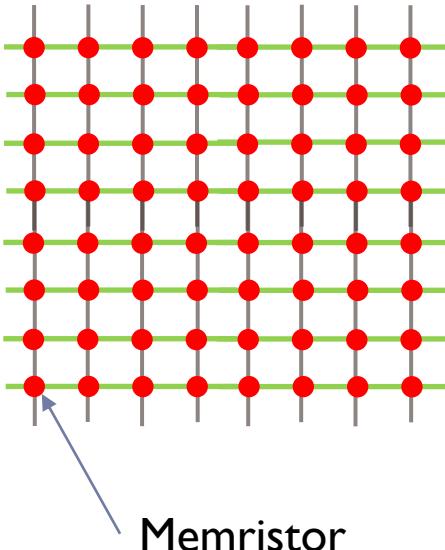
Incorrect read between a_1 and b_1 due to the alternate current path



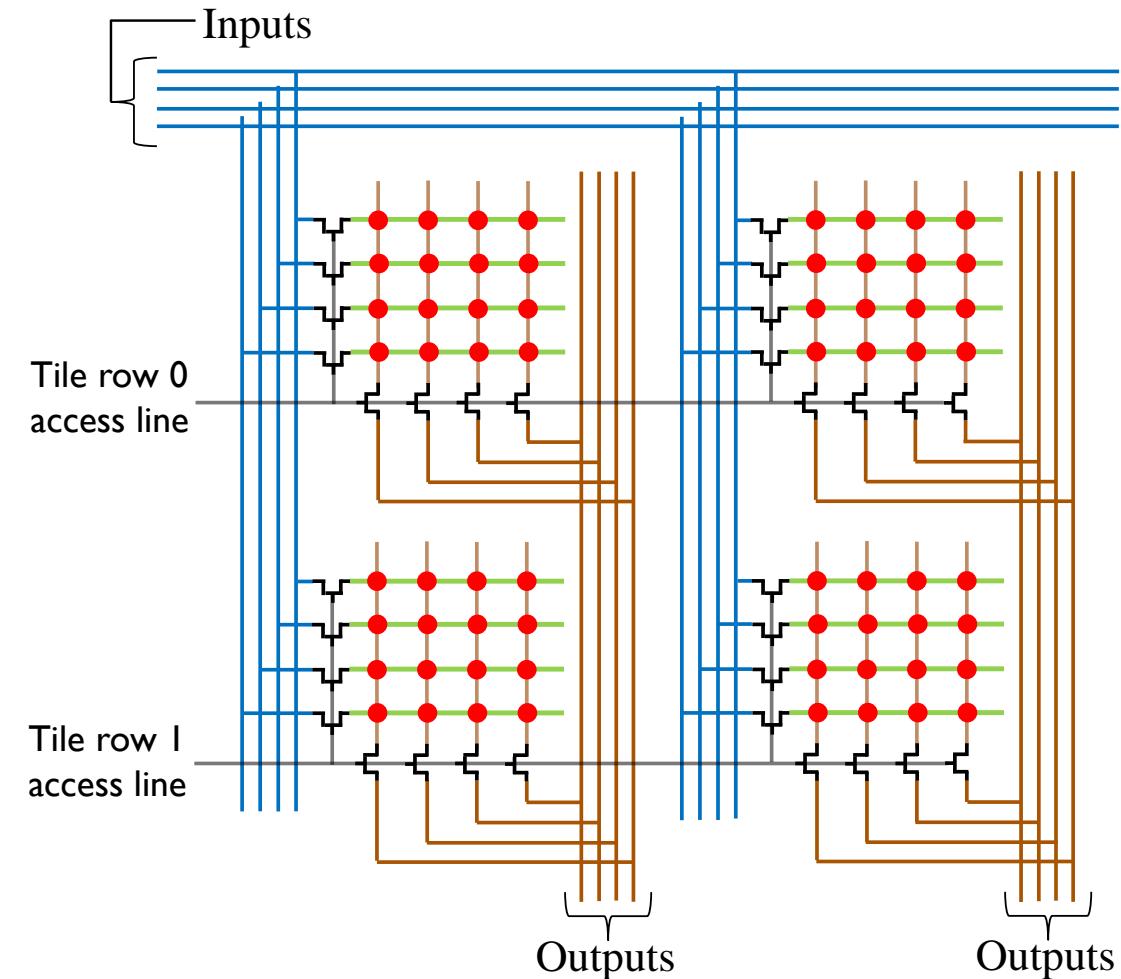
Tiled Memristor Digital Memory

- Full crossbar simulated in SPICE
- Wire resistances simulated
- Untiled memory array write energy very high
- Read noise margin low
- Designed a tiled memristor memory array

Untiled memory array



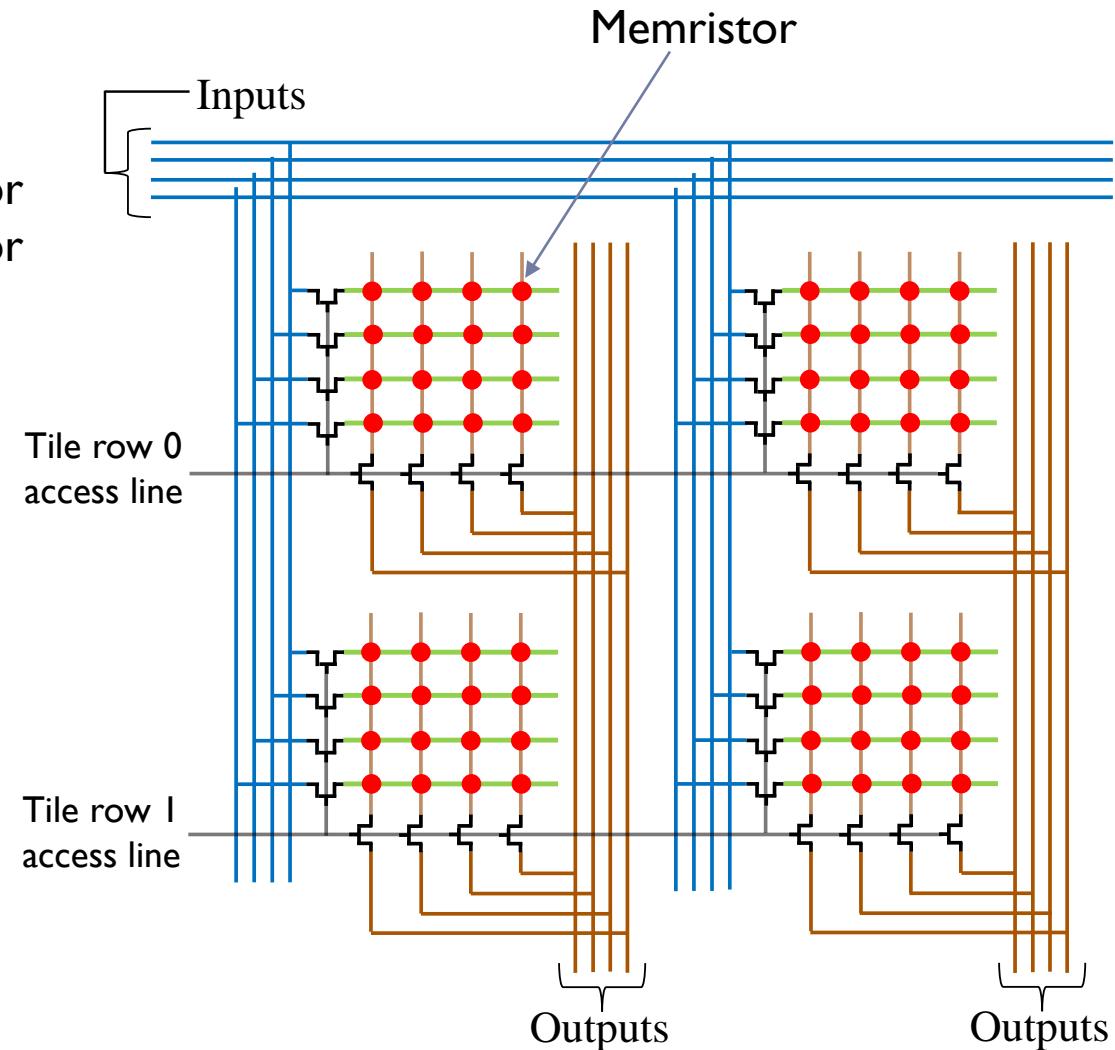
Tiled memory array



Memristor Memory Properties

- Training energy increases with crossbar size.
- Novel segmented memristor crossbar memory.
- 4x4 crossbar: 2 bits per transistor
- 8x8 crossbar: 4 bits per transistor

Crossbar size	Training energy per synapse	Read energy per synapse
4×4	1.90 pJ	1.84 fJ
8×8	5.07 pJ	2.00 fJ



Classifier Simulation

- ▶ 2 layer neural classifier
 - ▶ Based on a 16 memristor crossbar
 - ▶ Iteratively trained through MATLAB and SPICE

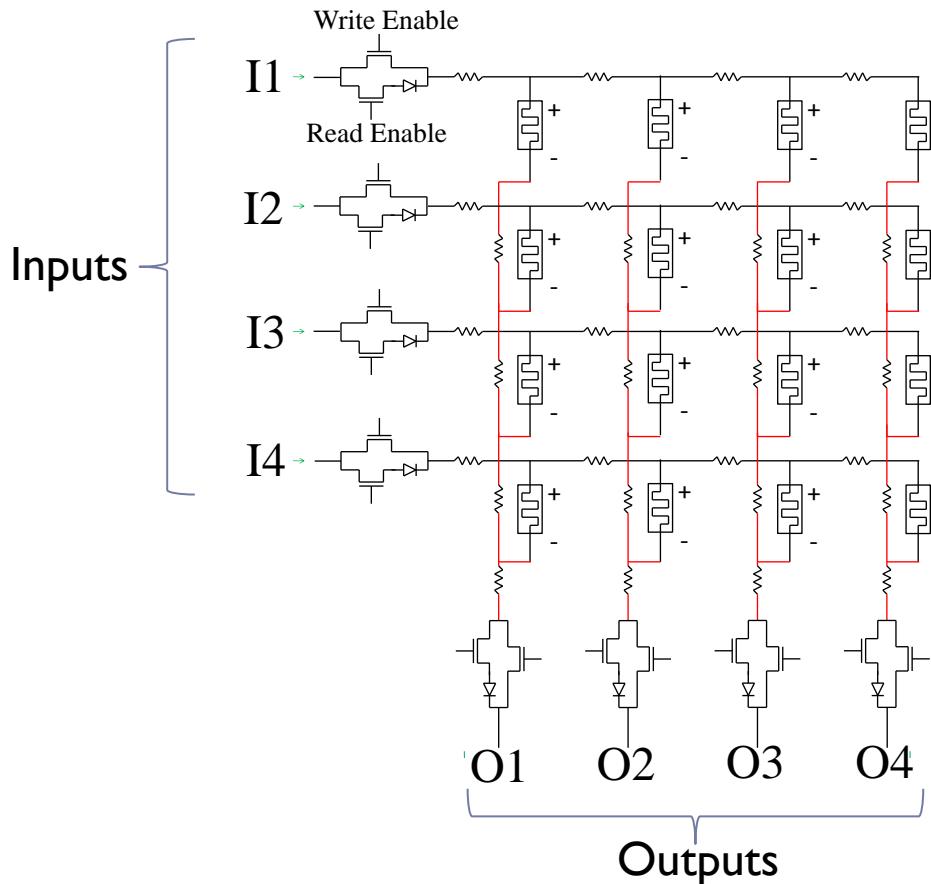
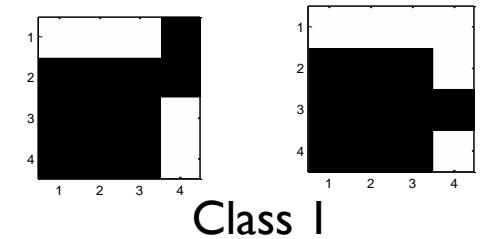
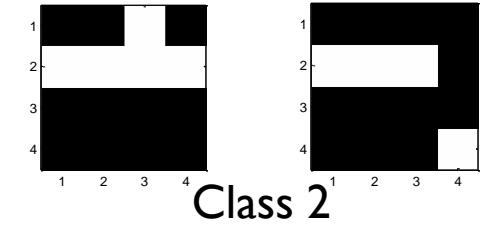


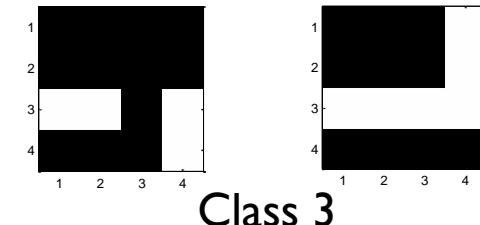
Image Classification Set



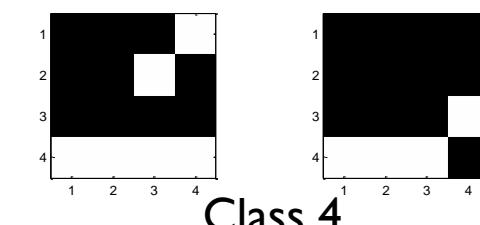
Class 1



Class 2



Class 3



Class 4

Simulation Setup

▶ Simulation platform

- ▶ Uses SPICE to produce accurate crossbar circuit simulation
- ▶ Output voltages and synaptic weights are analyzed in MATLAB

MATLAB / C

1. Generate weights and write to a SPICE configuration file
2. Call SPICE program with the new weight configuration file
3. Read weights from file
4. Simulate crossbar with different inputs and record outputs
5. Export output voltages to file
6. Evaluate output voltages from export file
7. Revise weights based on outputs
8. Go to step 2

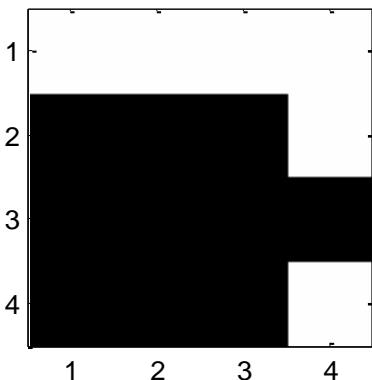


SPICE



Image Conversion

- ▶ Image data is converted to a SPICE waveform
- ▶ Each row in the image is converted to a voltage of 16 intervals
- ▶ Entire image is represented by 4 voltage pulses



1 1 1 1

0 0 0 1

0 0 0 0

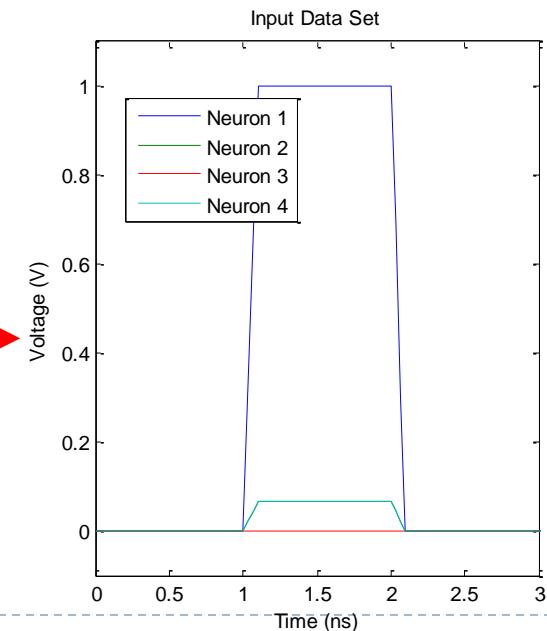
0 0 0 1

1V

1/15V

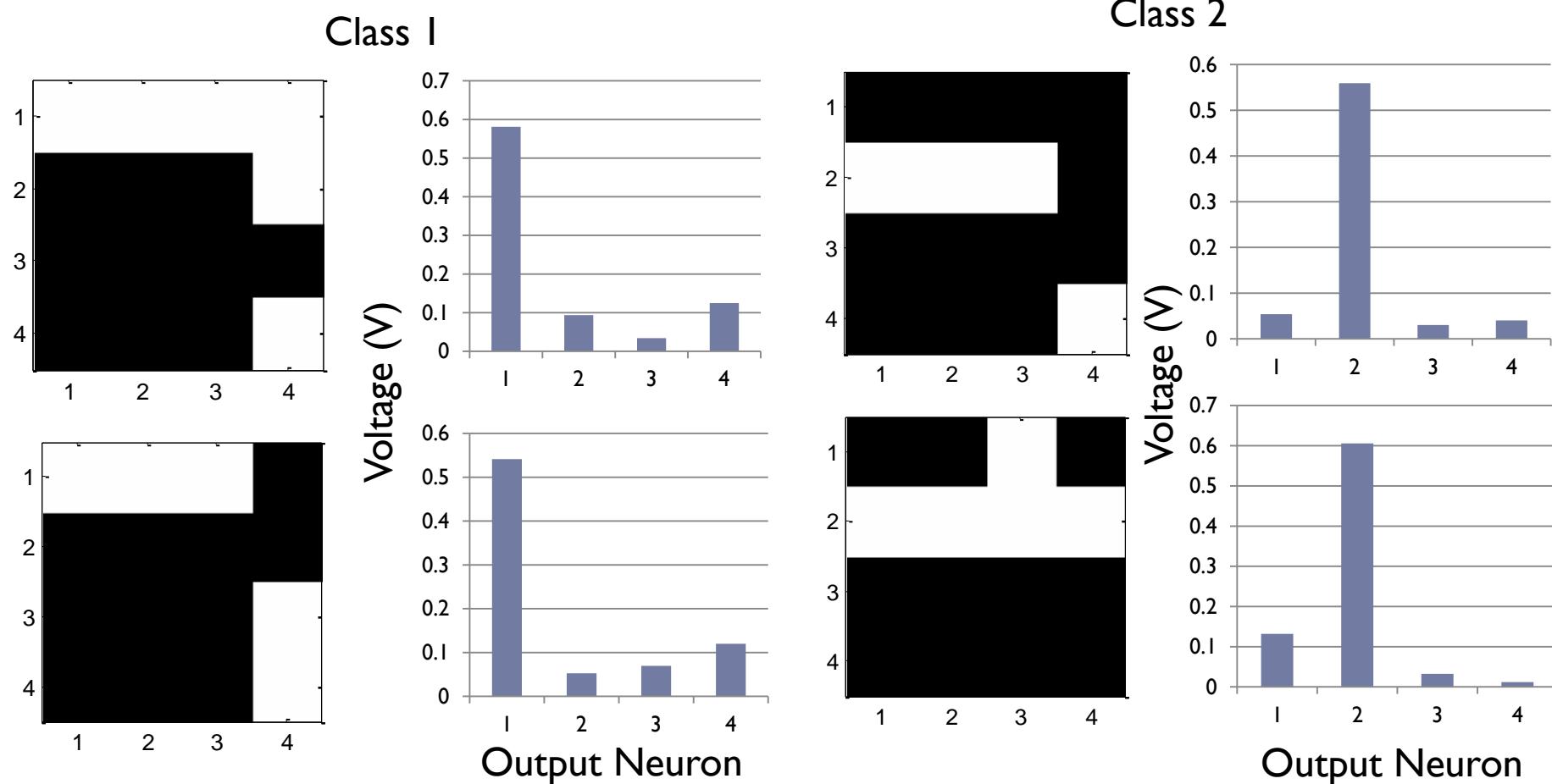
0V

1/15V



Simulation Result (1/2)

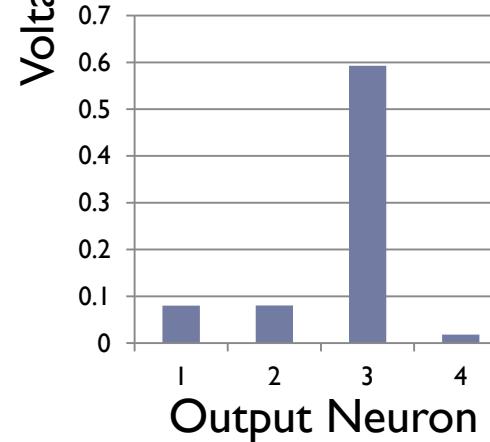
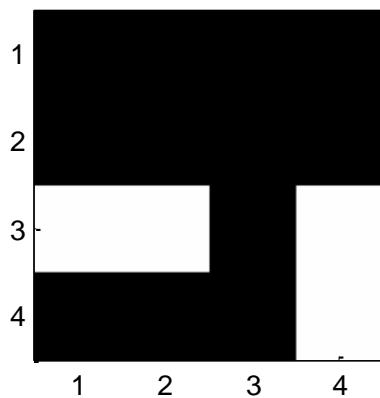
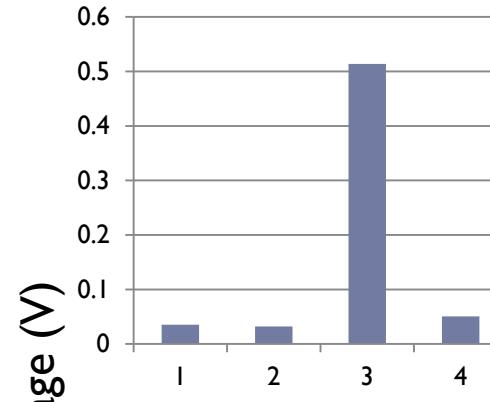
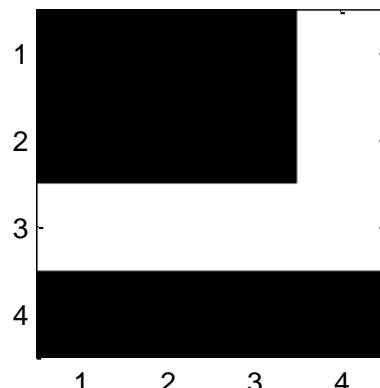
- ▶ Crossbar bar is performing as a linear separator
- ▶ Can differentiate between all 4 image classes



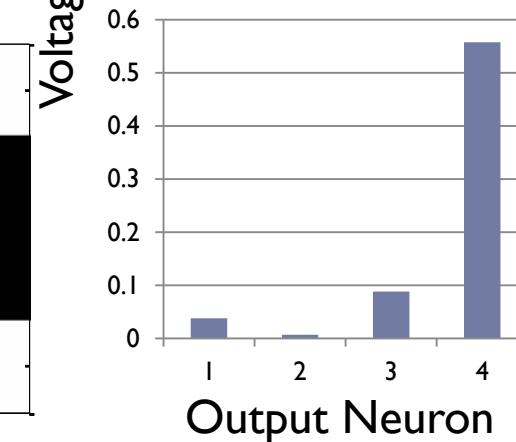
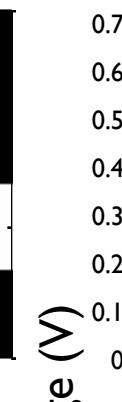
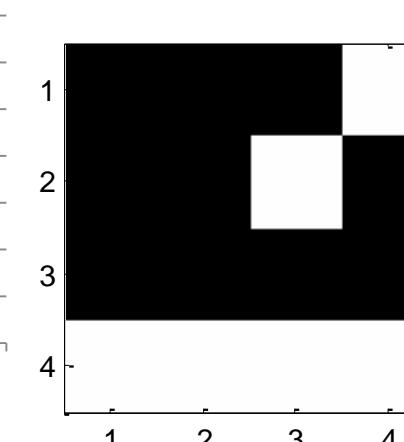
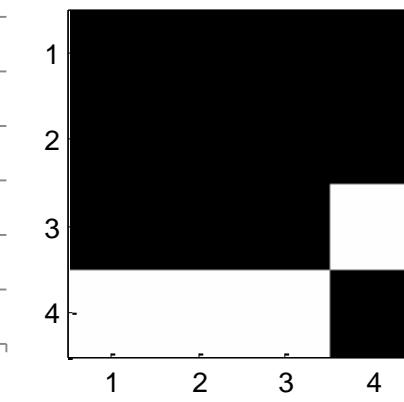
Simulation Result (2/2)

- ▶ Crossbar bar is performing as a linear separator
- ▶ Can differentiate between all 4 image classes

Class 3

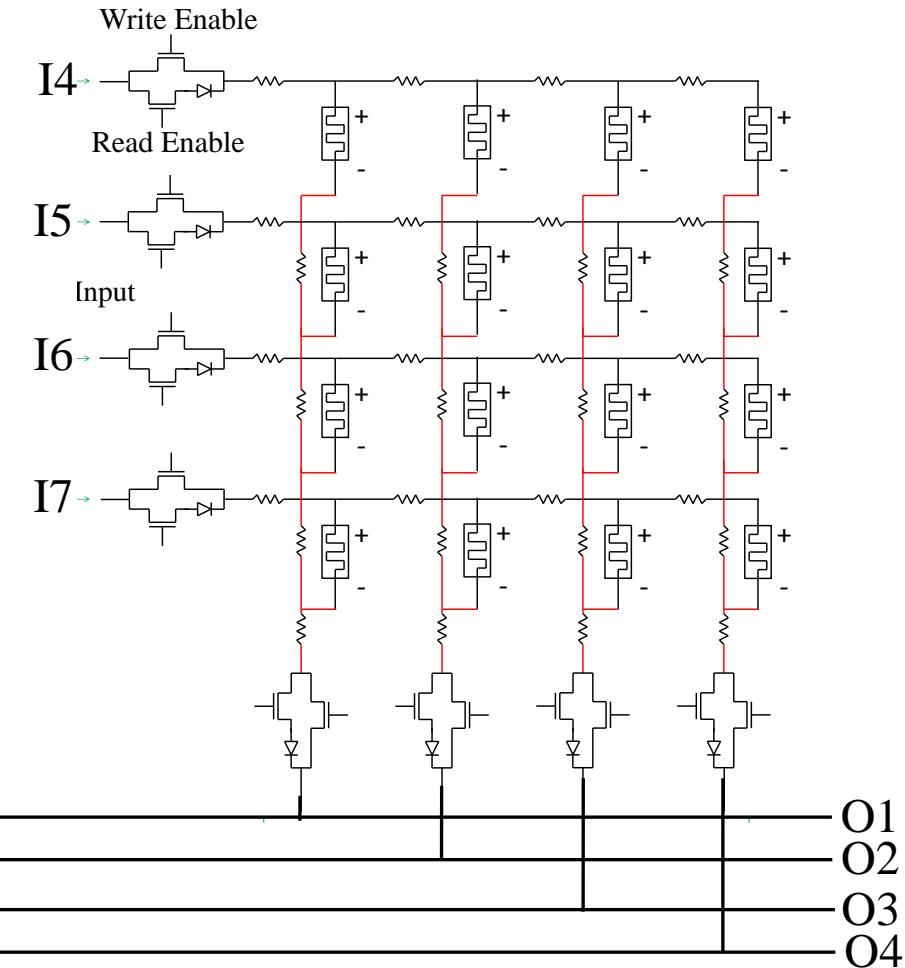
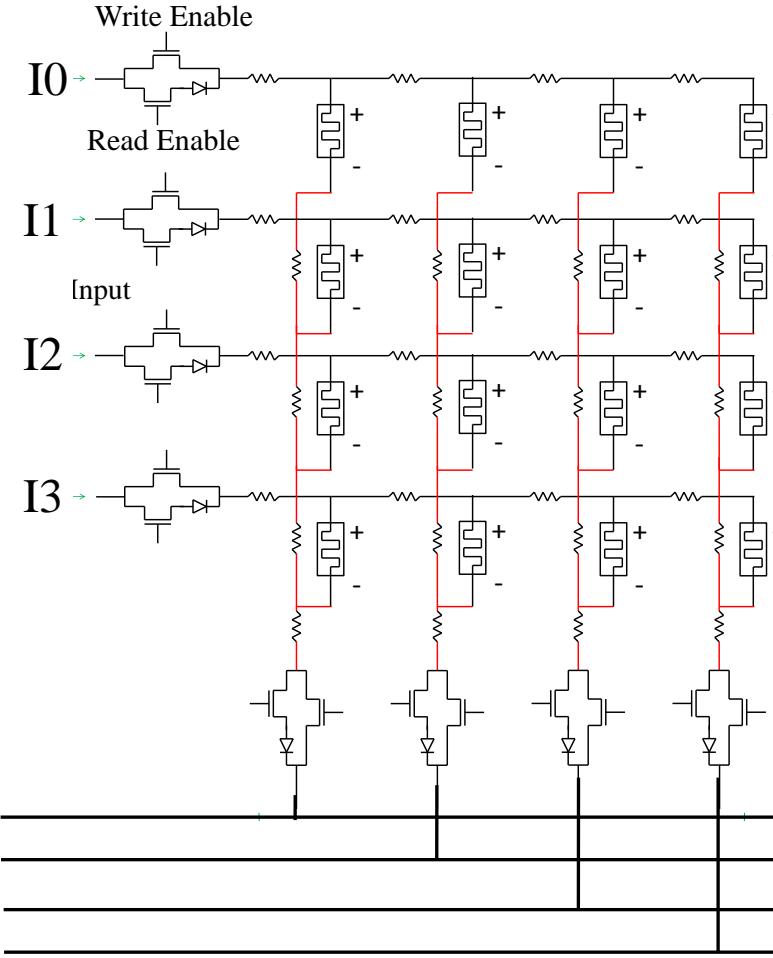


Class 4



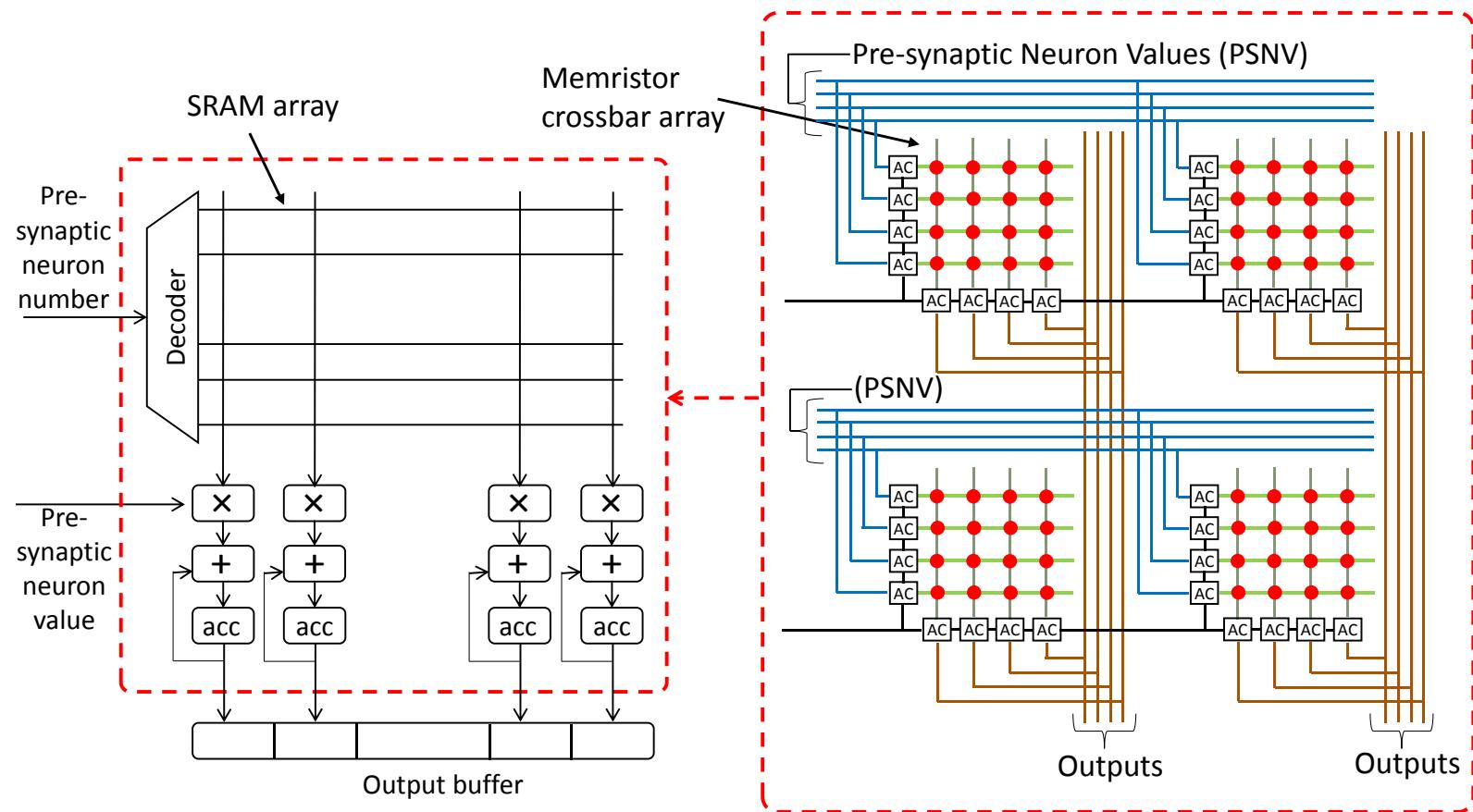
Parallel operation

- Read enable signal and diode allow multiple crossbars to be read in parallel
- Multipliers and adders not needed



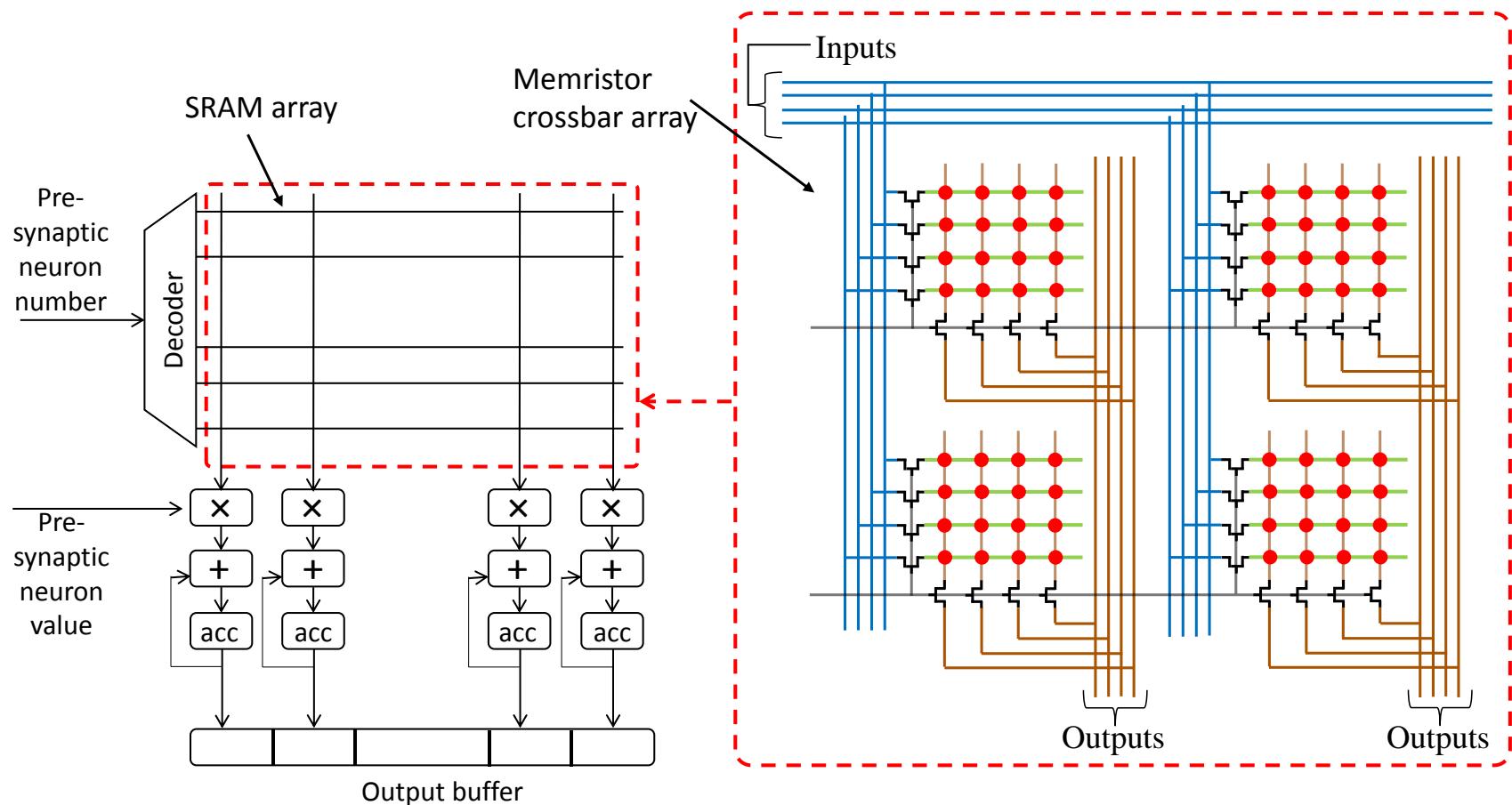
Analog Memristor Core

- Analog crossbar array replaces digital memory AND multiply-add units
 - Full crossbar can be evaluated in parallel
 - Most of the core can be shutdown once computations are done



Digital Memristor Core

- Digital crossbar array replaces SRAM memory array
- One row of synapses processed per cycle
- Most of the core can be shutdown once computations are done



Core Characteristics

Cycle Time: 5ns (200MHz clock)

Technology: 40nm

Cores for 1 bit per neuron case (multiplies not needed):

Config.	Synaptic memory device	Memory cells per synapse	Bits per synapse	Adder	Core area (mm ²)	Energy per neuron (pJ)	Core throughput (neurons/s)
1	Memristor	1	Analog	Current add	0.037	23	263.9×10^6
2	Memristor	2	2 bits	12 bit adder	0.098	240	42.1×10^6
3	SRAM	2	2 bits	12 bit adder	0.288	377	42.1×10^6

Cores for 4 bits per neuron case:

Config.	Synaptic memory device	Memory cells per synapse	Bits per synapse	Multiplier	Adder	Core area (mm ²)	Energy per neuron (pJ)	Core throughput (neurons/s)
4	Memristor	1	Analog	Memr. resist.	Current add	0.058	26	66.3×10^6
5	Memristor	4	4 bits	4 bit multiplier	18 bit adder	0.179	391	28.6×10^6
6	SRAM	4	4 bits	4 bit multiplier	18 bit adder	0.513	780	26.6×10^6

Energy considers leakage, data transfer to/from router, and control circuits.

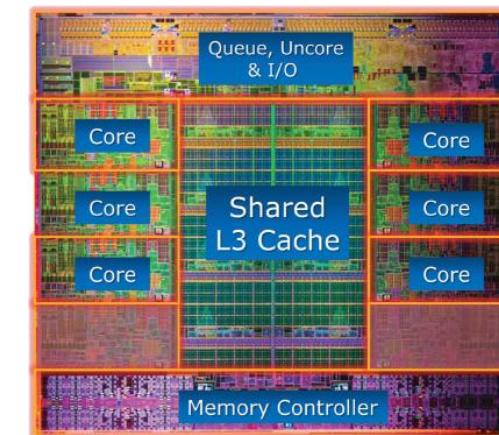
General Purpose Systems Comparison

- ▶ Programmed commercial systems:
 - ▶ Data set was small enough to fit in cache/onboard memory
 - ▶ Simulated neural networks with 1024 synapses per neuron
- ▶ NVIDIA Tesla M2070 GPGPU
 - ▶ 225W max power
 - ▶ 448 cores
 - ▶ 575 MHz
 - ▶ CUDA program
- ▶ Intel Xeon X5650 processor
 - ▶ 95W max power
 - ▶ Six cores
 - ▶ 2.66 GHz
 - ▶ SSE instructions modeled

NVIDIA GPGPU:



Intel Xeon:



Comparison: 1 Bit Neurons

Example 1:

- 25,600 neurons
- 100,000 iterations/s
- 1024 synap./neuron

Configuration	# of chips	Chip			Power (W)	Power eff. over Xeon
		area (mm ²)	% Active	%		
Memristor Analog	1	3.7	9.7%		0.07	253,489
Memristor Digital	1	9.7	60.8%		0.62	27,546
SRAM	1	35.2	60.8%		1.13	15,099
NVIDIA M2070	12	529.0	99.2%		2700.00	6
Intel Xeon X5650	179	240.0	99.9%		17005.00	1

Example 2:

- 1,706,667 neurons
- 1500 iterations/s
- 1024 synap./neuron

Configuration	# of chips	Chip			Power (W)	Power eff. over Xeon
		area (mm ²)	% Active	%		
Memristor Analog	2	248	0.15%		0.70	24,395
Memristor Digital	2	333	0.91%		1.25	13,633
SRAM	5	388	0.91%		28.02	607
NVIDIA M2070	12	529	99.2%		2700.00	6
Intel Xeon X5650	179	240	99.90%		17005.00	1

Comparison: 4 Bit Neurons

Example 1:

- 25,600 neurons
- 100,000 iterations/s
- 1024 synap./neuron

Configuration	# of chips	Chip area (mm ²)	% active	Power (W)	Power eff. over Xeon
Memristor Analog	1	5.9	38.6%	0.07	234,859
Memristor Digital	1	18.2	89.6%	0.62	16,968
SRAM	1	29.1	89.6%	1.13	8,215
NVIDIA M2070	12	529.0	99.2%	2700.00	6
Intel Xeon X5650	179	240.0	99.9%	17005.00	1

Example 2:

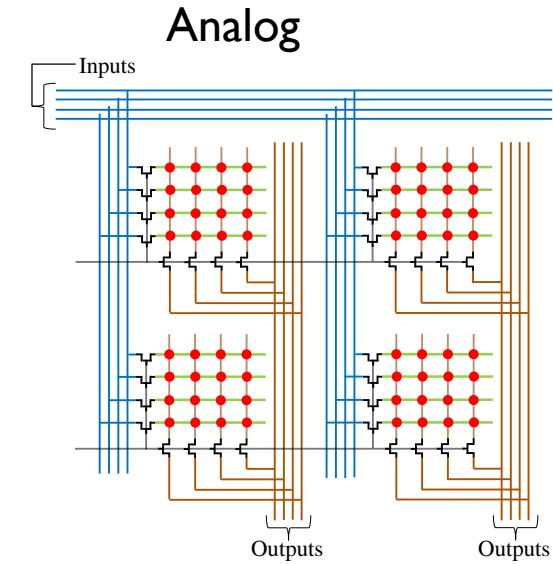
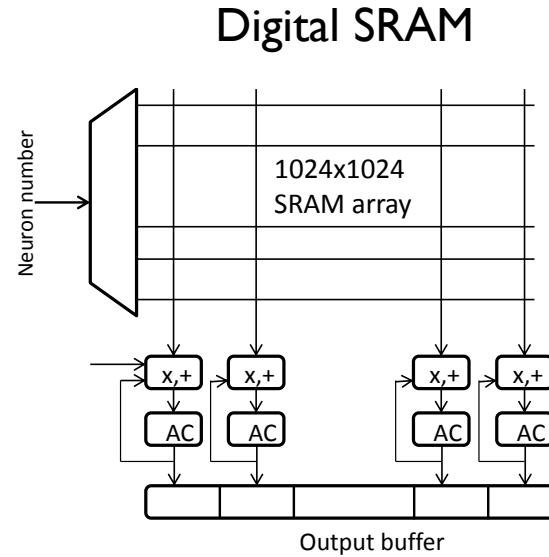
- 1,706,667 neurons
- 1500 iterations/s
- 1024 synap./neuron

Configuration	# of chips	Chip area (mm ²)	% active	Power (W)	Power eff. over Xeon
Memristor Analog	1	395	0.58%	0.70	24,210
Memristor Digital	4	303	1.34%	1.63	10,419
SRAM	9	383	1.34%	48.67	349
NVIDIA M2070	12	529	99.2%	2700.00	6
Intel Xeon X5650	179	240	99.9%	17005.00	1

Comparison: Digital vs Analog cores

256 Neurons with 1024 synapses each.

Synapses hold 16 possible values.



6x area $\rightarrow 29.1 \text{ mm}^2$

Add/Multiply \rightarrow Adder and multiplier needed

Slower \rightarrow 1 synapse processed /cycle (per neuron)

5.9 mm^2

Current summation

1024 synapses processed/cycle (per neuron)

0.07 W

16x more energy* $\rightarrow 1.13 \text{ W}$

*Energy calculation considers leakage and data routing power

Conclusion

- ▶ Specialized cores very efficient for neural network acceleration
 - ▶ Further improvements possible for both SRAM and memristor cores
 - ▶ Memristor cores efficient when carrying out recognition tasks
-
- ▶ Acknowledgement: This work was partially supported by an NSF Award

